CONFIGURABLE TERMINAL CAPABLE OF COMMUNICATING WITH VARIOUS REMOTE COMPUTERS

TECHNICAL FIELD

5   This invention relates to computer terminals, and more specifically, to an improved configurable computer terminal which can emulate a variety of different terminals and implement a variety of different communications protocols, depending upon which of a plurality of remote computers with which it is communicating.

10

BACKGROUND OF THE INVENTION

         Large main frame computers have been in use for many years. With the onset of personal computers however, many applications have moved towards being implemented as distributed systems. Specifically, many applications are now implemented by employing a plurality of personal computers on a network, each of which implements one or more functions required to implement an entire application or system.

20   Notwithstanding the widespread use of personal computers, the applications operating on large mainframe computers continue to be used.   Moreover, given the large transaction cost to convert systems running on large mainframes to PC based systems, it is unlikely that the large mainframe systems will be obsolete any time

25   soon.   Additionally, there are certain applications which may be implemented better on a large mainframe computer, such as an IBM S/390 or AS400.   Accordingly, many systems remain in use which utilize these large mainframes.

The large mainframes were initially designed to communicate with "dumb" terminals. However, recently it has become popular to utilize a personal computer instead of a dumb terminal, and to have the personal computer emulate the dumb terminal. Thus, the remote host believes it is communicating with a dumb terminal but it is actually communicating with a personal computer which presents the communications interface of a dumb terminal. Such a system allows the personal computer to interpret and reformat some of the information being transferred between the host computer and the terminal emulator so that a more user friendly interface is presented by the terminal emulator.

Most applications running on the host download to the terminal emulator a "green screen", a textual screen which is known to those in the computer industry and which is usually very user unfriendly. U.S. Patent No. 5,530,961 ("the '961 patent"), assigned to the same assignee as the present invention, describes a technique for identifying these green screens and presenting them in a graphical user interface (GUI) format. As taught by the '961 patent, the green screens are identified using one of a plurality of specific algorithms, and then the presentation of that screen is determined by a table look up.

U.S. patent application serial No. 08/722,583 ("the '583 application") also describes a plurality of screen recognition algorithms. Any of the algorithms and techniques set forth in the

2

'583 application or the '961 patent may be utilized to change a "green screen" to a more user friendly interface at a personal computer. Both the '961 patent and the '583 application are incorporated herein by reference.

5

While the '961 patent and the '583 application describe a technique which is a great improvement over the prior art, there is a potential storage and overhead problem presented by such arrangements. Specifically, there could be hundreds if not

10 thousands of possible different screens being downloaded from the host to the terminal. This results in an extremely large table having to be stored in each terminal. Moreover, most host computers communicate with many terminals and many terminals communicate with multiple hosts. Accordingly, each terminal

15 emulator must be able to maintain sufficient information to implement the communications software for a variety of different hosts, and to be able to identify and graphically present all of the screens which may be downloaded from each of the hosts.

20 SUMMARY OF THE INVENTION

A technical advance is achieved in accordance with the present invention which relates to a method and apparatus for providing a terminal emulator with the ability to emulate a variety of different terminals and to identify a plurality of different

25 screens. The invention is particularly applicable to thin clients, such as Network Computers (NCs), which are used in Internet

3

applications.

When the thin client desires to emulate the terminal and connect to a particular host, communications over a network are first established with a remote server, which downloads to the NC the appropriate communications software for communicating with a particular host. The communications software is then used to establish communications between the NC and the selected host.

Thereafter, as host screens are downloaded to the terminal emulator, the terminal emulator performs the screen recognition algorithm, and sends the results to the server. The server then returns to the terminal emulator the appropriate parameters for displaying the screen in GUI format.

In accordance with the foregoing, the terminal emulators may all be NC terminals since all of the tables and communications software are stored at a specified one or more servers. Moreover, it is not necessary to store all of the information required for the GUI screens at all of the terminals, as each can contact the server via, for example, the Internet.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 shows a conceptual block diagram of a terminal emulator connected to a host computer over a network for purposes of implementing the techniques of the present invention: and

5

Figure 2 shows a flow chart of exemplary software resident on a terminal emulator which can be used to implement the techniques of the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

Figure 1 shows a conceptual diagram of a network 101 and several computers and servers connected thereto. The network 101 is preferably the Internet, but may be any local area network (LAN) or wide area network (WAN). The network includes many thousands of computers, servers, and other similar items, several of which are shown for purposes of explanation.

The example of Figure 1 shows an NC terminal 104 which will be used for exemplary purposes herein. However, it should be understood that the invention is applicable to a variety of computing devices.

In operation, an exemplary NC terminal 104 is used to establish a communications session with a particular host 102. The host 102 is an IBM 3270 type of host which communicates with its terminals using a specified protocol, and includes a plurality of particular screens which it can download to terminals and which are dependent upon the particular applications actually running on host 102.

When the connection is desired to be made, NC terminal 104 of Figure 1 first connects with server 103 using standard Internet protocols. Server 103 includes various files of software which contain the appropriate communications software for communicating

5 with a variety of host computers, one of which is the IBM S/390 utilizing the TN3270 protocol for communications. The software for 3270 communications is downloaded to NC terminal 104 which may then utilize the software for communicating with host 102.

10 Additionally, the server 103 downloads to NC terminal 104 a table containing the unique identification numbers which will be generated by each screen downloaded from the host when the screen recognition algorithms of the '961 patent, or other such screen recognition algorithms, are executed. For each such unique

15 identification number, there is a 4 digit tag. An exemplary such table, denoted a screen table, is shown below.

Unique ID	Tag
123456	A002
579182	B013
613247	A005
421685	A042
683217	B401

Optionally, NC terminal 104 may send to server 103 information

25 regarding with which host it will be communicating. In such a scenario, server 103 may download only the communications software required for terminal 104 to communicate with host 102, and the appropriate screen tables. Alternatively, server 103 may download software and screen tables for all of the possible hosts, and

30 terminal 104 may then select the appropriate information to use.

6

During the implementation of a particular application, host 102 downloads a variety of different screens. These green screens, as they are called, are processed through screen recognition algorithms such as that described in the previously incorporated '961 patent and '583 application. The processing of each screen generates a unique identification number. The table previously downloaded to the NC is utilized to ascertain the tag from the unique identifier.

After the screen is processed, NC terminal 104 communicates with server 103. More specifically, the screen identifier is sent to server 103, and server 103 then locates the proper GUI interface information. The GUI information, which prescribes how to present the information in graphical form, is sent to NC terminal 104.

Figure 2 shows a flow chart of the software which may be resident in NC terminal 104 for implementing the configurable terminal in accordance with the techniques of the present invention. The flow chart is intended to show only the high level steps, as the programming details will be apparent to those of skill in this art.

After communications with server 103 is established at block 201, the communications software for communicating with a specified host is downloaded to NC terminal 104. Download screen ID 203 downloads the table of information from server 103 to terminal 104

for identifying the particular screen which has been downloaded from host 102. The connection from NC terminal 104 to host 102 is established at block 204, which implements a prescribed communication protocol.

5   Once the communications connection is established, screens of information, depending upon the particular application being run, will be downloaded from host 102 to NC terminal 104. These screens of information are waited for at block 205 by the NC terminal 204.

10   The screen of information is then placed into a presentation space and it is recognized using a screen recognition procedure at block 207. The particular screen recognition algorithm used is not critical to the present invention but may be of the type described in the '961 patent.

15

      After the screen is recognized and a screen ID is generated, the decision point 208 determines whether or not such screen is contained in the screen table at the server. If it is, then the tag is sent to the server and customization data is obtained from

20   the server at block 209. The customized GUI screen is presented to the user at block 210.

      On the other hand, if the list of screen IDs previously downloaded at block 203 does not contain the ID generated, then a

25   default GUI screen is presented. The default screen may be of any type desired by the user.

At block 212 a keyboard input, often including parameters entered into the GUI screen, is processed.

If there is a request for special services which may be resident only at server 103, then blocks 214 and 215 implement the functions of obtaining the special services and presenting them to the user. If there is no request for special services, then the keyboard input is simply processed in accordance with the application, and the algorithm returns to block 205 to await downloading of the next set of information from host 102.

Special services may include a variety of items dependant upon the screen and/or position of the cursor on the screen. For example, context sensitive help, lists of valid values for a field, display of signatures associated with the field, etc..

The special services may be downloaded in a manner similar to that utilized for downloading screens of information. Specifically, as previously explained, a tag is generated from the unique screen ID associated with each downloaded screen of information. The NC terminal utilizes the tag with a suffix in order to request that the GUI screen information be downloaded from the server. Thus, after tag B013, for example, is generated, a message indicating that the GUI screen information for screen B013 is being requested may be sent from the NC terminal to the server 103. This results in the server downloading the GUI screen

information to NC terminal 103.

One way of performing the foregoing is to simply send a message from NC terminal 104 to server 103 requesting B013.SCR, where the suffix SCR indicates that the screen information is requested.

When special services are being requested, such services may be dependent upon not only a screen of information at the NC terminal, but the location of the cursor on that screen. Thus, the aforementioned suffix may be replaced with a suffix indicating row and column number of the cursor. This would indicate the particular field where the cursor is, and thus an appropriate file at server 103 which contains information relevant to that field will be downloaded to NC terminal 104.

While the above describes the preferred embodiment of the invention, it is understood that various modifications and additions will be apparent to those of ordinary skill in the art. Such modifications are intended to be covered by the following claims.

10

11